所属类别	2021 年“华数杯”全国大学生数学建模竞赛	参赛编号

基于神经网络预测电动汽车目标客户销售策略

摘要

当今世界能源危机以及环境污染问题十分严重，而大力发展以电动汽车为代表的新能源汽车是解决上述问题的有效途径。由于受到多方面主客观因素的影响，消费者购买新能源汽车的意愿存在很大的不确定性。我们希望基于顾客满意度以及个人特征信息挖掘影响目标顾客是否愿意购买新能源汽车的主要因素，从而制定相应的销售策略。

问题 1 中，首先是数据清洗，对于异常值，利用**箱线图**找出 a1~a8 的离群异常值，并采用**均值**对原有数据进行修正，将 B17 中 300%这一数据，用**均值取整**后的数值替代。对于缺失值，将 B7 指标中的 NULL 用 0 代替。描述性分析中，本文采用**频数分布直方图**描述了 25 个指标的分布情况，并在第二问中分析了各指标与购车意愿的相关性。

问题 2 中，先将目标顾客个人特征因素 B1-B17 变量分为类别指标及数值指标：使用 Point-Biserial **相关性分析法**计算数值指标与购买意愿的相关性，最后总结对比得出**家庭可支配年收入**、**全年房贷**及**车贷支出占家庭年收入的比例**、**在本城市的居住**以及**工作时长**是影响他们购买意愿的主要因素；利用**卡方检验**计算类别指标与购买意愿间的关系，综合数据得知目标顾客的**学历**以及**单位性质**为主要影响因素。

问题 3 中，针对每个品牌的目标客户建立**多层感知机(BP 神经网络)预测模型**，对客户购买意愿进行预测。通过**正负样本均衡化**的方法使数据集两类样本的数量比例近似 1:1，以此提升网络的预测准确率。同时利用测试集实时进行检测，以判断数据集的拟合状态。最终模型在训练集和测试集上的准确率均在 80%以上，通过此模型预测出**第 1、5、6、7、12、13 个顾客**有购买意愿。

问题 4 中，以提升客户购买意愿值为目标，建立了**基于单目标非线性规划的销售策略模型**。首先将服务难度抽象为劳动成本，且劳动成本能够 1：1 地提升满意度。在成本总量一定的情况下，结合第三问建立的多层感知机预测模型，分别获得了**第 2、8、11 个顾客的最优销售策略**——分别应注重 $\mathbf{a_1}$、$\mathbf{a_2}$、$\mathbf{a_4}$，$\mathbf{a_3}$、$\mathbf{a_7}$、$\mathbf{a_2}$，$\mathbf{a_1}$、$\mathbf{a_6}$、$\mathbf{a_7}$ 满意度的提升。并且说服 3 位顾客购买汽车**至少需要劳动成本 7.09、37.9、45**。

最后，重新从各品牌待分析顾客中挑选另外 3 位制定销售策略，综合以上四问分析所得结果，给销售部门提出了针对每个品牌汽车的销售策略建议。针对品牌 1，应努力针对 B2、B15、B16、B17 等指标高的人群提升 $\mathbf{a_1}$、$\mathbf{a_2}$ 的服务质量；针对品牌 2，应努力针对 B13、B15、B16、B17 等指标高的人群提升 $\mathbf{a_2}$、$\mathbf{a_3}$、$\mathbf{a_7}$ 的服务质量；针对品牌 3，应努力针对 B10、B16、B17 等指标高的人群提升 $\mathbf{a_1}$、$\mathbf{a_6}$、$\mathbf{a_7}$ 的服务质量。

关键字：箱线图、Point-Biserial 相关性分析法、卡方检验、多层感知机预测模型、单目标非线性规划

一、 问题重述

1.1 问题背景

当今世界能源危机与环境污染问题日趋严重，受燃油费用、汽车尾气排放标准、国家扶持政策等各方面因素的影响，新能源汽车以其优良的工作性能与无污染的优点迅速成为汽车行业新的发展方向，市场前景广阔。与传统汽车相比，新能源电动汽车的某些领域如电池耐用性能等问题成为消费者是否愿意购买的参考因素，其市场销售急需科学决策。

为研究消费者对电动汽车的购买意愿，从而制定相应的销售策略，销售部门邀请1964 位目标客户对三种品牌（合资品牌、自主品牌、新势力品牌）电动汽车进行体验，得到了目标客户对新能源汽车不同方面的体验数据以及体验者个人特征信息。

1.2 问题提出

问题 1：做数据清洗工作，指出异常值和缺失数据以及处理方法。对数据做描述性统计分析，包括目标客户对于不同品牌汽车满意度的比较分析。

问题 2：结合电动汽车本身因素、目标客户个人特征因素等信息，研究哪些因素可能会对不同品牌电动汽车的销售有影响。

问题 3：建立不同品牌电动汽车的客户挖掘模型，并评价模型的优良性。并运用模型判断 15 名目标客户购买电动车的可能性。

问题 4：在附件 3 每个品牌中各挑选 1 名没有购买电动汽车的目标客户，实施销售策略。

问题 5：根据前面的研究结论，给销售部门提出不超过 500 字的销售策略建议。

二、问题分析

本文首先通过箱线图检验异常值并进行均值修正，通过 Point-Biserial 相关性计算与卡方检验分析得到影响客户购买意愿的指标。之后建立多层感知机模型并预测客户购买意愿。最后结合单目标非线性规划与多层感知机模型制定基于品牌的销售策略。

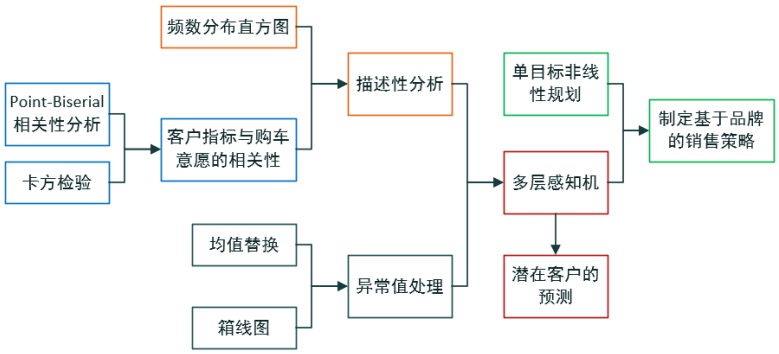


图 2-1 全文总体思路图

2.1 问题一分析

问题一要求对题目所给数据进行清洗工作以及对目标客户三种品牌的满意度进行

比较分析。

首先进行数据预处理。根据箱线图进行检测异常值，并利用均值修正。结合 B6 指标发现 B7 指标中 NULL 代表 0。做描述性分析时，B1~B17 指标需要分为类别指标与数量指标，分别做出各指标不同品牌的频率分布直方图；做 a1~a8 指标的频率分布直方图发现目标客户对品牌 1、2、3 的满意度依次递减。

2.2 问题二分析

问题二要求分析哪些影响因素会对客户购买不同品牌的意愿产生影响。

由于 B1~B17 分为类别指标与数值指标，我们需要分开分析。对于类别指标，由于因变量与自变量均为类别变量，所以我们利用 Pearson 卡方进行检测，若计算得到的概率值较大，则说明两变量之间相关性不大。对于数值指标，利用 Point-Biserial 相关性计算，对得到的结果进行排序可得到相关性影响。结合数值指标与分类指标整体来看可得到决定目标顾客购买新能源汽车的主要个人特征因素。

2.3 问题三分析

问题三要求基于前两问的研究成果建立客户挖掘模型，并对附件 3 的 15 名客户的购买意愿进行预测。

首先使用多层感知机（BP 神经网络）建立不同品牌的客户的挖掘模型，然后对客户的购买意愿进行预测。我们通过正负样本均衡化的方法使数据集两类样本的数量比例近似 1:1，以此提升网络的预测准确率，同时利用测试集实时进行检测，以判断数据集的拟合状态。

2.4 问题四分析

问题四要求通过提高客户满意度来提高其购买意愿，需要对不同的品牌制定不同的营销策略。

我们以提升客户购买意愿值为目标，建立基于单目标非线性规划的销售策略模型。首先将服务难度抽象为劳动成本，且劳动成本能够 1：1 地提升满意度。在成本总量一定的情况下，结合第三问建立的多层感知机预测模型，分别制定各自的营销策略。

2.4 问题五分析

问题五要求通过前几问的研究成果对销售部门提出不超过 500 字的建议。

综合前四问分析所得结果，针对每个品牌汽车给销售部门提出了不同的销售策略建议。针对品牌 1，应努力针对 B2、B15、B16、B17 等指标高的人群提升 a_1、a_2 的服务质量；针对品牌 2，应努力针对 B13、B15、B16、B17 等指标高的人群提升 a_2、a_3、a_7 的服务质量；针对品牌 3，应努力针对 B10、B16、B17 等指标高的人群提升 a_1、a_6、a_7 的服务质量。根据不同品牌的目标顾客和顾客的满意度结果有针对性地去对目标顾客进行服务，着重为其讲解、展示以及邀请体验，从而提升其满意度，增强购买意愿。

三、模型假设

1. 客户所填写的《目标客户个人特征调查表》中提供的数据真实可靠。
2. 销售部门所选择的 1964 名目标客户时具有一定随机性。
3. 顾客购买意愿仅与题目所提供的影响因素有关，没有受到其他因素影响。
4. 影响顾客购买意愿的各个因素相互独立，互不影响。

5. 工作人员可短时间内通过加强营销力度来提高目标客户的满意程度。

四、符号说明

符号	含义
a1~ a8	体验指标满意度得分
B1~B17	目标客户体验者个人的特征信息
$x_j^{(i)}$	第 i 个样本的第 j 个特征
$W^{[1]}$	输入层到第 1 层隐藏层的权值矩阵
$W^{[2]}$	第 1 层隐藏层到输出层的权值矩阵
$b^{[1]}$	第一层隐藏层的偏置矩阵
$b^{[2]}$	输出层的偏置矩阵
$z^{[1](i)}$	第 i 个样本作为输入时第 1 层隐藏层的未激活输出
$a^{[1](i)}$	第 i 个样本作为输入时第 1 层隐藏层的激活输出
$z^{[2](i)}$	第 i 个样本作为输入时输出层的未激活输出
$a^{[2](i)}$	第 i 个样本作为输入时输出层的激活输出(预测值)
α	更新速率(学习率)
a	类别 A(购买)样本数量
b	类别 B(未购买)样本数量
n	b 整除 a 的值
W	支付员工工资的资金（劳动成本）
x	满意度提高的百分点数

五、模型的建立与求解

5.1 数据预处理及满意度分析

5.1.1 数据预处理

1）缺失值处理

由于人为或者机械原因而导致的数据收集或保存失败从而造成的数据缺失，往往可能会使系统丢失大量有用信息，由此可见找出缺失值并采取相应措施显得尤为重要。

对附件进行清洗筛查时发现，B7 中存在 500 个 NULL 数据，B7 代表目标客户家庭中孩子的个数，结合指标 B6(目标客户的婚姻家庭情况)又发现，当 B7 为 NULL 时，B6 为“未婚”、“已婚无子女”、“离异/丧偶”、“其他”四种情况，即均无子女，所以可以确定 NULL 数据代表家庭中子女个数为 0，故将 B7 中的 NULL 全部用 0 替换。

2）异常值处理

异常值是指在数据集中存在的不合理的值，数据集中的异常值可能是由于人工录入错误或异常事件导致。如果忽视这些异常值，在某些建模场景下就会导致结论的错误，所以在数据预处理过程中，有必要识别出这些异常值并处理好它们。由于箱线图选取异常值比较客观，并且在识别异常值方面有一定的优越性，因此采用箱线图进行检测异常值。

为筛查满意度得分异常值，对 a1~a8 使用箱线图法[1]进行处理,但不同于经典箱线图法，本题由于数据分布较为分散，将上分位数取 90%分位数所对应的点 u_{90}，下分位

数取 10%百分位数所对应的点 u_{10}，数据中大于 u_{90} 或小于 u_{10} 的数据将被视作异常值，作箱线图如下所示：

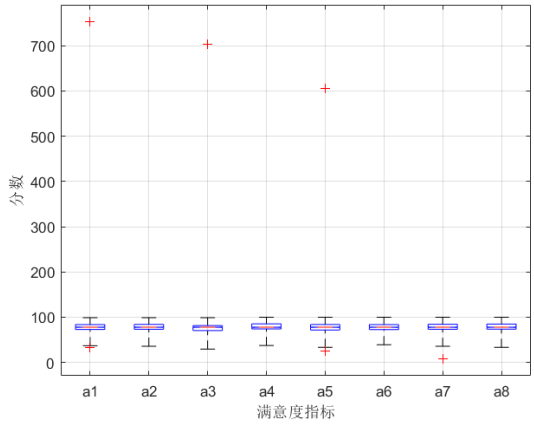


图 5-1 a1～a8 满意度指标箱线图

上图中红色的“+”代表的是异常值，需要将其剔除并利用均值加以修正。修正之后绘制 a1～a8 满意度的频数分布直方图如下所示：

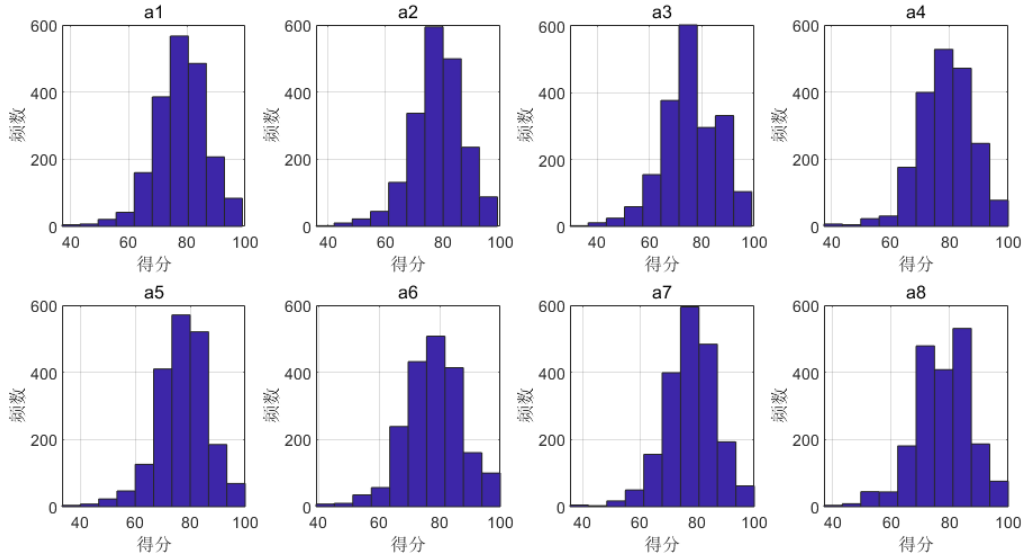


图 5-2 a1～a8 满意度频数分布直方图

同样对 B1~B17 的数据进行筛查，通过检查每一列数据是否符合该指标的取值要求来修正异常值，进而发现 B17(全年车贷的支出占家庭年总收入的比例)中出现 300%这一数据，显然有误，故以均值取整后的数值将原值进行替代。

5.1.2 满意度分析

由于品牌之间的差异性，我们需要对不同的品牌分别进行分析。在进行数据预处理之后，我们对不同品牌客户的各个满意度做柱形对比图，如下图 5-3 所示：

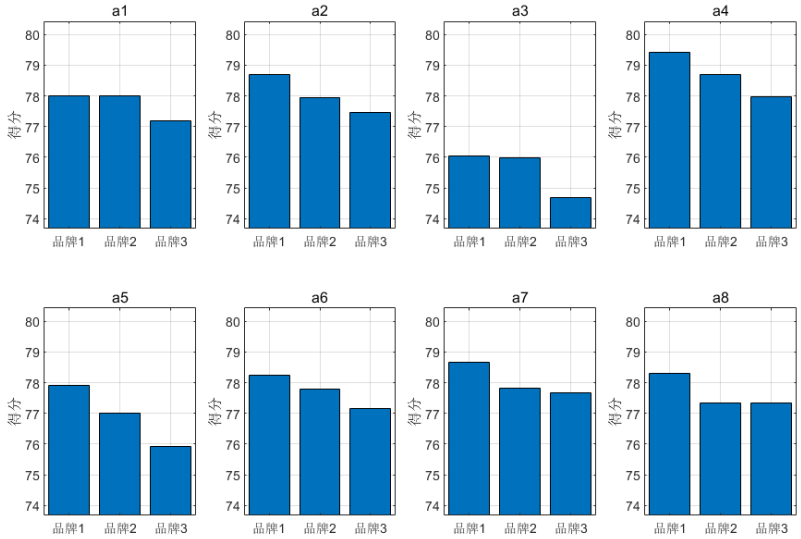


图 5-3 目标客户对于不同品牌满意度对比图

客户的满意度是从 a1~a8 这 8 各方面来进行展示的，由上图可以明显看出，客户对 3 种品牌汽车的满意度由高到低依次为品牌 1>品牌 2>品牌 3。

对于目标客户个人体验者特征信息 b1~b17，我们可将其分为两大类，类别指标与数值指标，如表 5-1 所示：

表 5-1 目标客户个人体验者特征信息分类表

类别指标	B1、B3、B6、B9、B11、B12
数值指标	B2、B4、B5、B7、B8、B10、B13、B14、B15、B16、B17

对于类别指标我们做频数分布直方图如图 5-4 所示：

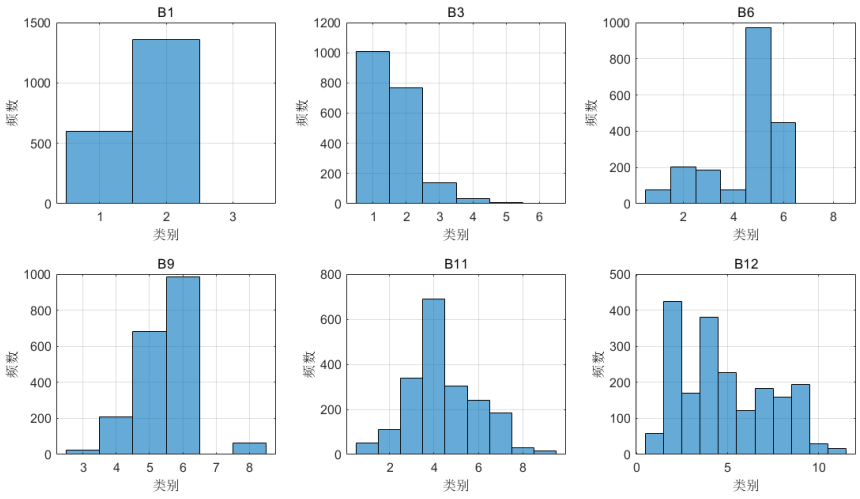


图 5-4 类别指标的频数分布直方图

从图中我们可以发现顾客大多居住在城市，并且已婚育有小孩，学历较高。

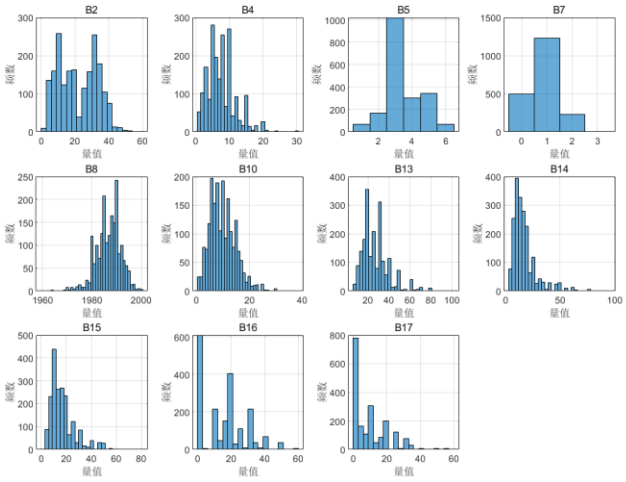


图 5-5 数值指标的频数分布直方图

从图中我们可以发现顾客在本城市居住时长大多在四十年以下，十年以下驾龄者居多，家中人口数多为 3 至 5 人，孩子不超过两个，基本均为 1980 年后生人，工作年限基本在 20 年以内，大多家庭收入在 40 万元以内，多数顾客个人年收入在 20 万元以内，多数家庭可支配年收入在 20 万元以内，车贷支出大多在 30%以内。

5.2 影响电动汽车销售因素

5.2.1 连续指标

连续数值变量与二分类变量的相关性可以使用 Point-Biserial 相关性系数[2]来衡量，其计算方式可以用下式来表示：

$$r_{pb} = \frac{M_1 - M_0}{s_n}\sqrt{pq} \tag{5-1}$$

其中，M_1代表二分类变量为“1”的均值，即购买意愿为“是”的均值，M_0代表二分类变量为“0”的均值，即购买意愿为“否”的均值，p 代表二分类变量为“1”所占的比例，即购买意愿为“是”的比例，q 代表二分类变量为“0”所占的比例，即购买意愿为“否”的比例，s_n为连续数值变量的标准偏差。计算目标顾客个人特征因素中的各连续变量（即数值指标）与购买意愿的相关性，将三个品牌分开计算得到如下表所示的结果：

表 5-2 数值指标与购买意愿的相关性表

品牌	B2	B4	B5	B7	B8	B10	B13	B14	B15	B16	B17
1	0.085	0.063	-0.005	0	0.014	0.011	0.035	0.046	0.086	-0.19	-0.17
2	0.057	-0.007	0.004	0.012	-0.013	-0.022	0.119	0.070	0.111	-0.21	-0.16
3	0.072	0.065	-0.017	0.008	-0.057	-0.097	-0.008	-0.038	0.063	-0.17	-0.08

其中绝对值越大代表相关性越强，正数代表正相关，负数代表负相关[3]。将以上所得数据进行比较，得出各数值指标与目标顾客的购买意愿相关性排序如下：

品牌 1：B16>B17>B15>B2>B4>B14>B13>B8>B10>B5>B7，即 B16、B17、B15 以及 B2 对品牌 1 的购买意愿的影响较大；

品牌 2：B16>B17>B13>B15>B14>B2>B10>B8>B7>B4>B5，即 B16、B17、B13 以及 B15 对品牌 2 的购买意愿影响较大；

品牌 3：B16>B10>B17>B2>B4>B15>B8>B14>B5>B13>B7，即 B16、B10 以及 B17 对品牌 3 的购买意愿影响较大。综合比对目标顾客对三种品牌的购买意愿可知，B15 家庭可支配年收入、B16 全年房贷支出占家庭年收入的比例、B17 全年车贷支出占家庭年总收入的比例、B2 顾客在本城市的居住时长以及 B10 工作年限是影响他们购买意愿的主要因素。

由于经济水平是决定目标顾客是否能够购买的关键因素，同时在本城市的工作以及居住时长也从侧面反映出他们对于出行工具的需求大小，时长越长的顾客在一定程度上对上下班通勤、短途郊游和接送老人孩子出发的需求越高，用户购买需求更加理性，更加注重综合表现，会从工作、家庭角度出发去选购新能源车。综合考虑到以上分析结果，与我们利用 Point-Biserial 相关性分析法计算目标顾客个人特征因素中的连续变量与购买意愿的相关性所得结果相一致。

5.2.2 分类指标

利用 Pearson 卡方检验[6]可从统计学的角度来描述二分类变量之间的相关性，即目标顾客个人特征因素中的分类变量与购买三种不同品牌汽车意愿间的关系，其计算公式如下：

$$\chi^2 = \Sigma \frac{(f_o - f_e)^2}{f_e} \tag{5-2}$$

其中，f_o代表实际频数，f_e代表期望频数，计算所得到的 Pearson 卡方越大则说明拒绝原变量之间的相关性越大，即代表两变量之间相关性较小。将三种品牌分开计算，得到结果如下：

表 5-3 分类指标与购买意愿的 Pearson 相关性表

品牌	B1	B3	B7	B9	B11	B12
1	0.878	0	0.83	0.703	0.189	0.545
2	0.420	0.461	0.705	0.012	0.001	0.035
3	0.789	0.612	0.861	0.312	0.684	0.712

由上表数据可知目标顾客对品牌 1 的购买意愿与 B3 居住区域、B11 所在单位性质具有较大的相关性；目标顾客对品牌 2 的购买意愿与 B9 最高学历、B11 单位性质以及 B12 职位具有较大的相关性；目标顾客对品牌 3 的购买意愿与 B9 最高学历具有一定相关性。综合以上结果可知，目标顾客对各品牌新能源汽车的购买意愿与他们的学历以及单位性质有一定程度上联系。

结合数值指标与分类指标整体来看，决定目标顾客购买新能源汽车的个人特征因素主要是经济实力、相应需求以及受教育程度。

5.3 基于多层感知机的客户挖掘模型

在问题 2 的基础上，我们使用多层感知机（BP 神经网络）建立不同品牌的客户的挖掘模型，然后对客户的购买意愿进行预测。由于是针对不同品牌电动车分别进行预测，所以多层感知机模型也是在不同品牌的训练集下进行训练的，故一共要训练 3 个多层感知机模型 $f_1(x)$、$f_2(x)$、$f_3(x)$，分别进行品牌 1、2、3 的客户购买意愿预测。

5.3.1 单层感知机模型

多层感知机是基于单层感知机的网络结构[4]。现搭建如下单隐藏层（4 个神经元）神经网络，并介绍感知机的预测机理：

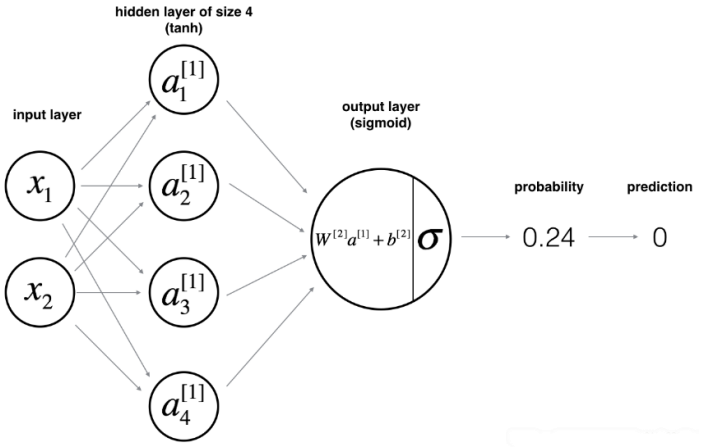


图 5-6 单隐藏层神经网络结构图[5]

这里的数学符号与前文符号说明中代表的意义一致。

1）前向传播

对于某个输入样本 $x^{(i)}$，前向传播过程可以表示为如下几个公式[9]：

$$\begin{aligned} z^{[1](i)} &= W^{[1]}x^{(i)} + b^{[1]} \\ a^{[1](i)} &= \tanh(z^{[1](i)}) \\ z^{[2](i)} &= W^{[2]}x^{(i)} + b^{[2]} \\ \hat{y}^{(i)} &= a^{[2](i)} = sigmoid(z^{[2](i)}) \end{aligned} \tag{5-3}$$

其中 sigmoid 激活函数和 tanh 激活函数表达式[9]如下：

$$\begin{aligned} sigmoid: f(z) &= \frac{1}{1+e^{-z}} \\ \tanh: f(z) &= \frac{2}{1+e^{-2z}} - 1 \end{aligned} \tag{5-4}$$

最后一层的激活函数使用 sigmoid 的原因是要把输出值限制在 0~1 之间，作为判断事物类别的概率依据。

2）损失函数

损失函数使用的是交叉熵损失[9]：

$$J = -\frac{1}{m}\sum_{i=0}^{m}(y^{(i)}\log(a^{[2](i)}) + (1-y^{(i)})\log(1-a^{[2](i)})) \tag{5-5}$$

3）反向传播

反向传播的计算公式如下:

$$
\begin{aligned}
dz^{[2]} &= a^{[2]} - y \\
dW^{[2]} &= dz^{[2]} a^{[1]^T} \\
db^{[2]} &= dz^{[2]} \\
dz^{[1]} &= W^{[2]^T} dz^{[2]} * g^{[1]'}(z^{[1]}) \\
dW^{[1]} &= dz^{[1]} x^T \\
db^{[1]} &= dz^{[1]}
\end{aligned}
\tag{5-6}
$$

其中为了计算 $g'(z^{[1]})$，因为使用了 tanh 激活函数那么假设 $a = g(z)$，那么 $g'(z) = 1 - a^2$。对于 sigmoid 激活函数同样假设 $a = g(z)$，那么 $g'(z) = a(1-a)$。

4）参数更新

更新算法即梯度下降法，如下所示：

$$
\theta = \theta - \alpha \frac{\partial J}{\partial \theta} \tag{5-7}
$$

其中 α 是更新速率(学习率)。

5）模型整合

构建神经网络的一般办法是：

(1) 定义神经网络结构：

输入单元数量：2，隐藏单元数量：4

(2) 初始化模型参数 $W^{[1]}, b^{[1]}, W^{[2]}, b^{[1]}$

使用 matlab 指令 rand 生成[-0.005,0.005] 之间的随机数。

(3) 循环(当不超过最大迭代次数时)：

A. 实施前向传播

B. 计算损失

C. 实施后向传播

D. 更新参数(梯度下降)

算法流程图如下所示：

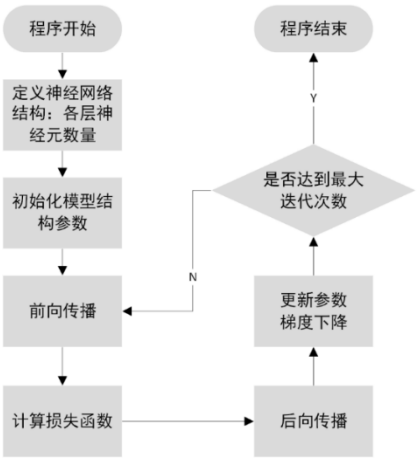


图 5-7 单层感知机分类器的算法流程

5.3.2 多层感知机模型

本题使用的多层感知器模型是基于上述单层感知器模型的，共两个隐藏层，每个隐藏层 4 个神经元，输入的数据具有 25 个特征（a1~a8，B1~B17），并直接将类别变量对应的类别量化处理。双隐藏层神经网络结构图如下所示：

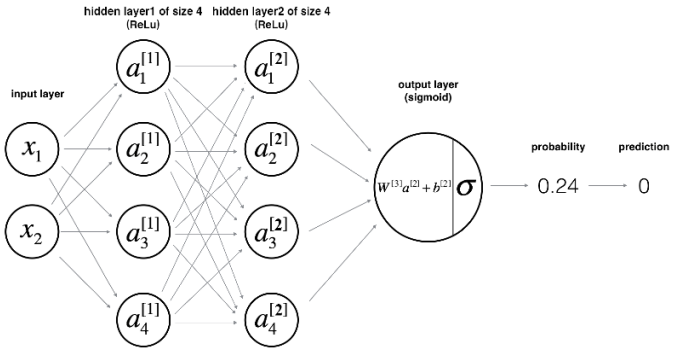


图 5-8 双隐藏层神经网络结构图

双隐藏层神经网络的参数初始化、前向传播、损失函数计算和参数更新具体实现步骤与单隐藏层几乎一致，在中间层的激活函数使用的是 ReLu 激活函数，这里给出从最后一层输出层到第一层隐藏层的逐层反向传播公式，其中大写字母代表的都是矩阵。

$$
\begin{cases}
dA^{[3]} = \dfrac{1-Y}{1-A^{[3]}} - \dfrac{Y}{A^{[3]}} \\
dZ^{[3]} = dA^{[3]} * \dfrac{1}{1+e^{-Z^{[3]}}} * (1 - \dfrac{1}{1+e^{-Z^{[3]}}}) \\
dW^{[3]} = \dfrac{1}{N} dZ^{[3]} \cdot A^{[2]} \\
db^{[3]} = \dfrac{1}{N} \sum_{i=1}^{N} dZ^{[3](i)} \\
dA^{[2]} = W^{[3]^T} \cdot dZ^{[3]} \\
dZ_i^{[2]} = \begin{cases} dA_i^{[2]} & Z_i^{[2]} > 0 \\ 0 & Z_i^{[2]} \le 0 \end{cases} (i = 1,2.....,n_2) \\
dW^{[2]} = \dfrac{1}{N} dZ^{[2]} \cdot A^{[1]} \\
db^{[2]} = \dfrac{1}{N} \sum_{i=1}^{N} dZ^{[2](i)} \\
dA^{[1]} = W^{[2]^T} \cdot dZ^{[2]} \\
dZ_i^{[1]} = \begin{cases} dA_i^{[1]} & Z_i^{[1]} > 0 \\ 0 & Z_i^{[1]} \le 0 \end{cases} (i = 1,2.....,n_1) \\
dW^{[1]} = \dfrac{1}{N} dZ^{[1]} \cdot X \\
db^{[1]} = \dfrac{1}{N} \sum_{i=1}^{N} dZ^{[1](i)} \dfrac{1}{1+e^{-Z^{[3]}}})
\end{cases}
\tag{5-8}
$$

反向传播的流程与之前单隐藏层的类似，需要注意的是其中*表示数乘，· 表示矩阵乘法。n_1 是第一层隐藏层的神经元数目，n_2 是第二层隐藏层的神经元数目。

5.3.3 多层感知机模型的求解

1）训练测试数据预处理

如果输入的客户特征数据不进行归一化，神经网络训练效果十分糟糕，分类器容易失效。分别对客户的每一个特征进行归一化，归一化方式：

$$X^* = \frac{X - \mu}{\sigma} \tag{5-9}$$

为了确定训练集和测试集，需要对数据集进行分割，且要保证训练集中，每一类样本所占比例近似（1:1），否则神经网络更倾向于将输入样本全部预测为训练集中较多的那类样本。

首先对数据集的分布进行一个宏观的展示：

表 5-4 原数据集正负样本比例

	购买汽车的人数	未购买汽车的人数	购买：未购买
品牌 1	23	533	0.0432
品牌 2	65	1208	0.0538
品牌 3	11	124	0.0887

显然，两类样本分布极其不均衡，购买汽车的样本数数远远小于未购买汽车的样本数。这里采用复制购买汽车的客户样本的方法来扩充数据集，然后再将数据集打乱，能够一定程度上缓解正负样本不均衡的问题。

具体算法流程如下：

设 a 为类别 A(购买)的样本数量,b 为类别 B(未购买)的样本数量。

$$n = \frac{[b - \mathrm{mod}(a,b)]}{a} \tag{5-10}$$

n 是 b 整除 a 的运算结果。将类别 A 的样本复制 n+1 次，于是类别 A 就有 $a(n+1)$ 个样本，且 $a(n+1) \geq b$。将类别 A 的后 $a(n+1)-b$ 个样本删除，此时类别 A 和类别 B 的样本数目为 1：1，实现了均衡化。实践证明，将样本均衡化能够极大提高网络的预测准确率。

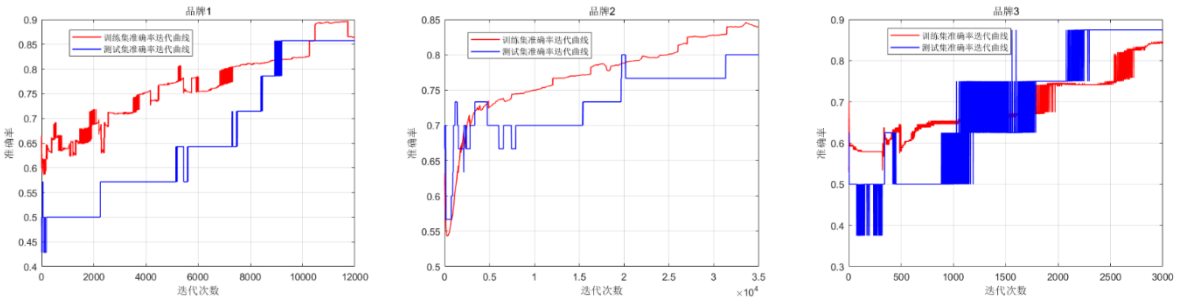
为了避免过拟合和欠拟合现象的发生，除了使用训练集进行训练之外还需要使用测试集实时进行检测，用以判断数据集的拟合状态，原数据集正负样本比例表如下所示：

表 5-5 原数据集正负样本比例表

	训练集		测试集	
	购买	未购买	购买	未购买
品牌 1	526	526	7	7
品牌 2	1193	1193	15	15
品牌 3	120	120	4	4

3）模型求解结果

下面分别是三种品牌的神经网络各自训练过程中准确率的迭代曲线：



（a）品牌 1　　（b）品牌 2　　（3）品牌 3

图 5-9 三种品牌准确率迭代曲线图

由此看出训练结果还是较为优良，并没有出现过拟合以及欠拟合现象，模型收敛到一个比较好的状态。最终模型在训练集和测试集上的准确率如下表所示：

表 5-6 模型在训练集和测试集的预测准确率

	训练集准确率	测试集准确率
品牌 1	86.4068%	85.7143%
品牌 2	83.9899%	80.0000%
品牌 3	84.5833%	87.5000%

由上表可以看出，训练集与测试集的准确率较高，可以依据此来进行预测。然后对目标客户进行预测，得到第 1 个客户(品牌 1)、第 5 个客户(品牌 1)、第 6 个客户(品牌 2)、第 7 个客户(品牌 2)、第 12 个客户(品牌 3)和第 13 个客户(品牌 3)有购买新能源汽车的意愿。

5.4 基于单目标非线性规划的销售策略模型

5.4.1 建立模型

本问是一个单目标多约束数学规划问题，关键是要找到优化问题的三要素：决策变量，目标函数，约束条件[7]。

假定服务难度与工资相关，即工作人员的服务难度越高，老板需要付出的工资越多，我们假设总的可用于支付员工工资的资金为 W 元，现在需要根据资金进行分配给各个影响因素；又由于服务难度与提高满意度的百分点是成正比的，满意度提高的百分点数用 x 表示，故：

$$W = kx \tag{5-11}$$

不妨假设 k=1，且 x_i 表示第 i 个满意度指标的百分点提升值，所以原问题转化为如何将 W 分配给 x_i（i=1,2,3,...,8）。使得客户购买该品牌汽车的意愿最强烈。

所以 x_i 可以视作决策变量，一共有 8 个决策变量。

目标函数是为了衡量客户购买该品牌汽车的意愿程度，在这里可以使用第三问中训练好的多层感知机模型 $f(a_1,a_2,...,a_8,B_1,B_2,...,B_{17})$ 来获得顾客的意愿值。在多层感知机模型中，模型输出值是神经网络输出层 sigmoid 函数的输出值，故模型的值域为(0,1)，即：

$$f(a_1,a_2,...,a_8,B_1,B_2,...,B_{17}) \in [0,1] \tag{5-12}$$

且当顾客意愿值大于 0.5 时，意味着顾客愿意购买汽车。所以本问的目标决策函数 $g(x_1,x_2,...,x_8)$ 可以表示为：

$$g(x_1,x_2,...,x_8)=f(a_1+x_1,a_2+x_2,...,a_8+x_8,B_1,B_2,...,B_{17}) \quad (5\text{-}13)$$

因为是针对某一客户的销售策略，所以 $a_1,a_2,...,a_8,B_1,B_2,...,B_{17}$ 这 25 个参数是已知的。决策函数的值越大意味着顾客购买该品牌汽车的几率越大。

同时由于 $f(a_1+x_1,a_2+x_2,...,a_8+x_8,B_1,B_2,...,B_{17})$ 的前 8 个输入量必须小于 100，因为满意度得分最高为 100，所以决策变量满足下面的约束：

$$x_i<100-a_i \quad (i=1,2,..,8) \quad (5\text{-}14)$$

而且 x_i 是由 W 分配得到，故需要满足：

$$\sum_{i=1}^{8}x_i\le W \quad (5\text{-}15)$$

所以需要使用销售策略在 W 一定的情况下，让顾客最容易购买汽车。于是该单目标多线性约束数学规划模型可以表述如下：

$$\max \quad g(x_1,x_2,...,x_8)=f(a_1+x_1,a_2+x_2,...,a_8+x_8,B_1,B_2,...,B_{17})$$

$$s.t.\begin{cases} x_i<100-a_i \quad (i=1,2,..,8) \\ \sum_{i=1}^{8}x_i\le W \end{cases} \quad (5\text{-}16)$$

5.4.2 模型的求解

从附件 3 每个品牌中随机挑选 1 名没有购买电动汽车的目标客户如下表：

表 5-7 未购买电动汽车的目标客户

客户编号	品牌编号	a1	a2	a3	a4	a5	a6	a7	a8
2	1	89.84	87.65	88.92	88.88	88.87	91.63	99.99	99.98
8	2	82.80	83.24	77.81	85.61	84.02	83.60	82.15	81.09
11	3	88.80	82.63	81.56	81.03	80.84	75.03	80.05	74.44

显然当劳动成本 W=0 时，即不使用任何销售策略的时候，客户都是不愿意购买的，即

$$g_j(0,0,...,0)<0.5 \quad (j=1,2,3) \quad (5\text{-}17)$$

其中，j 代表的是某品牌，虽然每个品牌有着相同多层感知器模型结构，但模型参数有所不同，所以针对每个品牌的目标客户，销售策略应有所不同。下面以品牌 1 的客户为例，当劳动成本 W 取某一固定值，比如 W=5 时，使用 Matlab 的 fmincon 函数求解该非线性规划模型，得到 $g(x_1,x_2,...,x_8)$ 局部最优值，以及决策变量 x_i 的取值情况如下表所示：

表 5-8 W=5 时客户 2 的最优销售策略

W	x1	x2	x3	x4	x5	x6	x7	x8	最优意愿值
5	5	0	0	0	0	0	0	0	0.499999

由于 x1=W，即应该全部投入到 x1 中，所以当 W=5 时，应该专门针对电池技术性能提升服务质量。也就是说电池技术性能应该放在首位考虑。此时顾客的意愿值小于 0.5，说明仍然不愿意购买。在仅投入 W=5 的服务量的情况下不能满足需求。

分别计算当 W=10、20、30、40、60、80、100 时的销售策略得：

表 5-9 W 不同时客户 2 的最优销售策略

W	x1	x2	x3	x4	x5	x6	x7	x8	最优意愿值	$\sum_{i=1}^{8} x_i$
10	10	0	0	0	0	0	0	0	0.500000782	10
20	14.95	5.04	0	0	0	0	0	0	0.500002059	20
30	14.95	12.34	0	2.70	0	0	0	0	0.500003481	30
40	14.95	12.34	1.57	11.12	0	0	0	0	0.500004611	40
60	14.95	12.34	18.44	11.12	0	0	3.13	0	0.500006173	60
80	14.95	12.34	18.44	11.12	0	0	15.68	0	0.500006418	72.56
100	14.95	12.34	18.44	11.12	0	0	15.68	0	0.500006418	72.56

接下来对表格进行分析，当 W=20 时，x_1=14.95,此时结合表 1，得 a_1+x_1=100,不能再增加 x_1，因为第一项满意度已经达到满分。这时根据寻优结果显示，应该增加第二项满意度，一直直到第二项满意度达到满分，再增加第四项满意度。往后同理。可以看出当 W 逐渐充足时，销售策略的重点依次为 $a_1 > a_2 > a_4 > a_3 > a_7$。

同时可以看出当 W>72.56 后，第 1、2、3、4、7 项满意度指标均达到 100，但此时就算 W 有剩余，也不分配给第 5、6、8 项指标，经过验证，如果将 W 剩余的量分配完全,反而会降低顾客的最优意愿值，不利于顾客购买该车。

为了更全面地探索顾客意愿值的变化情况，在最优销售策略的指导下，将 W 的取值进一步细分，取间隔为 1，得到顾客意愿值的变化情况如下图：

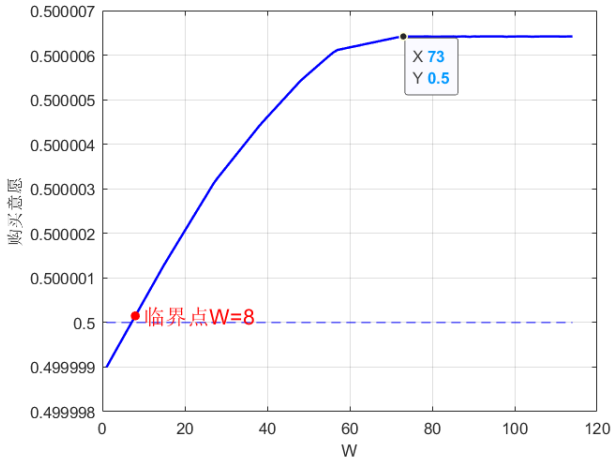


图 5-10 顾客 2 购买品牌 1 汽车的意愿值变化情况

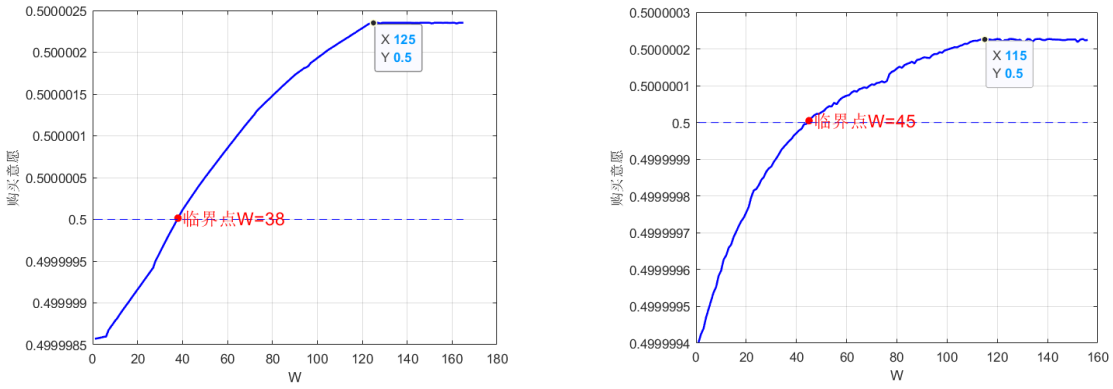
从上图可以看出，当劳动成本 W=8 时，$g(x_1,x_2,...,x_8)>0.5$，便能够说服顾客购买品牌 1 的汽车。当劳动成本 W 增大到 73 左右时，便能使顾客的购车意愿达到最大。所以说服 2 号顾客购买汽车的最低成本为 73，销售策略如下：

表 5-10 顾客 2 的最佳销售策略

劳动成本 W	x1	x2	x3	x4	x5	x6	x7	x8
7.09	7.09	0	0	0	0	0	0	0

所以从上表可以看出，只需要耗费 W=7.09 的劳动成本，全部投入到第 1 项满意度（电池的技术性能）的提升上即能说服顾客 2 购买品牌 1 的汽车。

同理可以求出其余两位顾客的最佳销售策略，首先将 W 值进一步细分，分别做出两个顾客意愿值的变化情况：



（a）顾客 8　　（b）顾客 11

图 5-11 顾客购买品牌 3 汽车意愿变化情况

分别当 W=38 和 W=45 时可以说服该两位顾客购买汽车。得到两位顾客的最佳销售策略：

表 5-11 顾客 8 的最佳销售策略

劳动成本 W	x1	x2	x3	x4	x5	x6	x7	x8
37.9	0	0	20.05	0	0	0	17.84	0

从表 5-11 中可以看出要让 8 号顾客购买汽车，要着重提高第 3 项（经济性）和第 7 项（外观内饰整体表现）满意度。

表 5-12 顾客 11 的最佳销售策略

劳动成本 W	x1	x2	x3	x4	x5	x6	x7	x8
45	11.20	0.31	0.02	0	0	15.06	18.39	0

从表 5-12 中可以看出要让 11 号顾客购买汽车，要着重提高第 1 项（电池的技术性能）、第 6 项（驾驶操控性表现）、第 7 项（外观内饰整体表现）满意度。

综合来看，下表给出了在销售过程中针对这三位顾客应当注重的指标重要性的排序：

表 5-13 三位顾客指标重要性排序

客户编号	品牌编号	指标重要性排序
2	1	$a_1 > a_2 > a_4 > a_3 > a_7 > a_5 = a_6 = a_8$
8	2	$a_3 > a_7 > a_2 > a_6 > a_1 > a_4 > a_8 > a_5$
11	3	$a_1 > a_6 > a_7 > a_5 > a_3 > a_2 > a_8 > a_4$

5.5 销售策略建议

企业最重要的是要去培养销售人才、服务人才、售后人才以及网络人才等，为扩大市场打下基础。对于重点目标客户群体设立专业部门，在国家较大的基建项目、旅游项目、游乐项目以及公共配套项目中进行推广，同时也要在大城市的重点小区去推广使用新能源汽车。

销售部门应进一步规范服务标准，通过加强对员工的培训，提高工作人员整体素质，提供先进的服务设施、提升和完善维修服务质量，提供完善的保险以及信贷业务等售后服务，从而使顾客享有更好的购物体验。随着经济的快速发展，生活水平迅速提升，人们对环境的要求也逐渐提高，由此可见在销售推广的过程中可以着重介绍其节能环保、使用成本低，经济效益较高等特色。同时提出国家大力支持新能源汽车产业的发展，并对购买者予以补贴，进一步增强吸引力。

根据不同消费者的需求，鼓励顾客参与体验，有效激发其潜在需求以及价值诉求，从而制定更加灵活有效的市场推广策略，充分尊重消费者意见，建立完善的营销机制。

除传统的新车试驾体验外，额外增加新能源发动机、新技术体验以及售后体验、维修体验等，拉近与消费者间的距离，满足不同消费需求。同时只依靠传统营销的模式远远不够，销售部门应充分借助新媒体以及社交网络进行宣传推广，进一步扩大年轻消费群体等。针对品牌 1，应努力针对 B2、B15、B16、B17 等指标高的人群提升 a_1、a_2 的服务质量；针对品牌 2，应努力针对 B13、B15、B16、B17 等指标高的人群提升 a_2、a_3、a_7 的服务质量；针对品牌 3，应努力针对 B10、B16、B17 等指标高的人群提升 a_1、a_6、a_7 的服务质量。

六、模型检验

6.1 数据集分割方式不同的影响

因为神经网络的训练是基于大量的数据，如果训练集和测试集的选取不同，得到的网络模型也不同[8]。由于本题训练样本中有意愿购买汽车的样本数为少数，通过重复使用这些样本，虽能一定程度上缓解正负样本分布不均的问题，但仍不能完全消除。

现将训练集或测试集中重复使用的样本除去，剩下的不愿意购买的样本为最原始的真实样本，是独一无二的，最具有参考价值。因为愿意购买汽车的人是少数，这部分样本的特征对于模型预测是极其重要的，所以现在要讨论这种真实样本在训练集和测试集中的数量分布情况对神经网络预测效果的影响。

由于测试集的用处是判断训练过程中神经网络是否出现过拟合和欠拟合现象，以下的测试都保证了神经网络未出现过拟合和欠拟合。

三种品牌各自愿意购买汽车的样本数分别是 23 人、65 人、11 人。现在将这些样本分配到测试集和训练集，然后观察最终预测结果是否不同。

表 6-1 数据集分割方式对预测结果的影响

	实验序号	测试集(人数)	训练集(人数)	预测结果
品牌 1	1	3	20	[1 1 0 0 1]
	2	4	19	[1 1 0 0 1]
	3	5	18	[1 1 0 0 1]
	4	6	17	[1 1 0 0 1]
	5	7	16	[1 0 0 0 1]
	6	8	15	[1 0 0 0 1]
	7	9	14	[1 0 0 0 1]
品牌 2	1	5	60	[1 1 0 0 0]
	2	10	55	[1 1 0 0 0]
	3	15	50	[1 1 0 0 0]
	4	20	45	[1 1 0 0 0]
	5	25	40	[1 1 0 0 0]
	6	30	35	[1 1 0 0 0]
	7	35	30	[1 1 0 0 0]
品牌 3	1	1	10	[0 1 1 0 0]
	2	2	9	[0 1 1 0 0]
	3	3	8	[0 1 1 0 0]
	4	4	7	[0 1 1 0 0]
	5	5	6	[0 1 1 0 0]
	6	6	5	[0 1 1 0 0]
	7	7	4	[0 1 1 0 0]

从上述表格可以看出，在品牌 1 的客户预测中，测试集和训练集的真实愿意购买的样本分配情况会对最终的预测造成一定影响，但影响很小，主要是第 2 号顾客的预测结果会被影响，因为愿意购买车辆的客户远远少于不愿意购买的客户，认为第 2 号顾客应当被判为不愿意购买。或者认为 2 号顾客处于不愿意与愿意购买的临界状态。

在品牌 2 和品牌 3 的客户预测中，数据集的分割方式并未对预测结果造成影响。综上所述，说明本题训练得到的多层感知机模型具有很好的鲁棒性，对数据集的分割方式不灵敏。

6.2 对不同顾客的分析

当顾客不同时，消费策略也需要重新进行制定。我们随机再从品牌 1~3 中各选择一人，利用神经网络进行优化，确定不同的分配系数，得到的结果如下所示。

表 6-2 顾客的最佳销售策略

顾客	劳动成本 W	x1	x2	x3	x4	x5	x6	x7	x8
3	75.15	23.71	25.38	26.04	0	0	0	0	0
9	11.46	0	0	4.73	0	0	0	6.72	0
11	122.10	23.11	14.03	25.94	0	24.69	22.22	12.07	0

然后逐渐增加成本 W，并对每个客户的销售策略进行分析，可以得出以下指标重要性的排序结果。

表 6-3 三位顾客指标重要性排序

客户编号	品牌编号	指标重要性排序
3	1	$a_1 > a_2 > a_3 > a_4 > a_7 > a_8 > a_6 > a_5$
9	2	$a_7 > a_3 > a_2 > a_6 > a_1 > a_4 > a_8 > a_5$
14	3	$a_1 > a_7 > a_6 > a_5 > a_8 > a_3 > a_2 > a_4$

将客户 3 与客户 2，客户 9 与客户 8，客户 14 与客户 11 的指标重要性排序进行比较，发现大体类似，所以对于每个品牌的顾客有不同的销售策略着重点，而且也说明了本模型具有很好的鲁棒性。

七、模型的评价

7.1 模型的优点

- 本文的模型是在对大量数据进行分析和处理后建立的，并且建模过程严格按照观察数据、发现问题-分析问题、提出假设-建立模型、验证模型-综合分析-规律总结与推广的科学探索过程来对问题进行研讨。
- 通过将数据描述性统计化可以方便直观地观察分析，帮助我们直观、快捷地寻找到数据间的关系，寻找普适规律，使模型建立的数据信息更加可靠，更加贴近实际。在定性的描述之后进行定量的计算，使结果更加可靠。
- 通过对模型的层层检验和比较，使模型更加可靠的同时能适应更加复杂的实际情况，模型简洁实用，可移植性强。

7.2 模型的缺点

- 在确定各变量对最终购买意愿的影响时，未考虑各变量之间的相互影响，仅计算其对购买意愿的影响。
- 现实生活中，对于服务质量人们往往很难定量地提升，无法做到模型中的精确计算。

7.3 模型的改进

- 在分析各因素变量对购买意愿的影响时，考虑各因素变量的内在相互影响。
- 使用更加优良的样本均衡化手段以及数据增强手段扩充有限数据集。
- 参数更新方式采用 Adam 优化算法的模型预测效果更为优良。
- 销售策略模型可以修正为更为复杂的双目标非线性规划模型，不仅追求提升顾客的购物意愿值，还追求降低服务难度，减少劳动成本。

7.4 模型的推广

- 该模型对于其他销售情况下分析同样具有很好的评估和预测作用，并对于商家制定销售策略起到一定的正向作用。
- 模型建立时对模型反复的检验和修正过程，同样可以迁移到其他数学模型的检验和修正。

参考文献

[1] 汪发余,高振沧, 毕建武. 基于 SPSS 组合预测算法的煤炭消费量预测研究 [J]. 资源开发与市场,2014,30(08):957-960.

[2] 王月辉, 王青. 北京居民新能源汽车购买意向影响因素——基于 TAM 和 TPB 整合模型的研究[J]. 中国管理科学, 2013.

[3] 田宗博, 承前. 消费者新能源汽车购买意愿影响因素分析——基于TPB理论和probit模型的研究[J]. 中国市场, 2017(22).

[4] Kemery E R , Dunlap W P , Griffeth R W . Correction for Variance Restriction in Point-Biserial Correlations[J]. Journal of Applied Psychology, 1988, 73(4):688-692.

[5] Bass A R , Ager J. Correcting Point-Biserial Turnover Correlations for Comparative Analysis[J]. Journal of Applied Psychology, 1991, 76(4):595-598.

[6] 黄代新, 杨庆恩. 卡方检验和精确检验在 HWE 检验中的应用[J]. 法医学杂志, 2004, 020(002):116-119.

[7] 李萍, 曾令可, 税安泽,等. 基于 MATLAB 的 BP 神经网络预测系统的设计[J]. 计算机应用与软件, 2008, 025(004):149-150.

[8] 苏高利, 邓芳萍. 论基于 MATLAB 语言的 BP 神经网络的改进算法[J]. 科技通报, 2003(02):45-50.

[9] Kulbear. Building your Deep Neural Network - Step by Step. https://github.com/Kulbear/deep-learning-coursera.(2017)

附录

第一问 数据预处理代码 data_preprocess.m

```
clc;clear;close;
load origidata.mat;
var(A,0,1);
A1=[];
A2=[];
for i=1:size(A,2)
    ai=A(:,i);
    N=size(A,1);
    q_=prctile(ai,[10,90]);
    p10=q_(1);
    p90=q_(2);
    upper=p90+1.5*(p90-p10);
    lower=p10-1.5*(p90-p10);
    upper_indexes=find(ai>upper);
    lower_indexes=find(ai<lower);
    indexes=[upper_indexes,lower_indexes];

    aver_ai=mean(ai(find(ai<upper&ai>lower)));
    ai(indexes)=aver_ai;
    A2=[A2,ai];

end
figure(2)
boxplot(A,'Notch','on','Labels',{'a1','a2','a3','a4','a5','a6','a7','a8'},'Whisker',3.7)
grid on
xlabel('满意度指标');
ylabel('分数');
%基本上都比较正常 只是 a7 有一个值比较小 是 7 把它替换成均值
%下面画一下 A 的分布图
%
figure(1)
for i=1:size(A2,2)
    ai=A2(:,i);
    subplot(2,4,i);
    hist(ai);
    ylabel('频数');
    xlabel('得分');
    title(['a',num2str(i)]);
    grid on;
end

%%把 null 全部换成 0
```

```
b7=B(:,7);
b7(find(isnan(b7)))=0;
B(:,7)=b7;
%%把 B17 中的离群值去掉
figure(3)
b17=B(:,17);
boxplot(b17,'Notch','on','Whisker',3.7);
b17(b17==300)=mean(b17(b17~=300));
B(:,17)=b17;

data=[brand,A2,B,will];
s = xlswrite('data.xls', data);
```

第一问 描述性分析& 第二问 Point serial 相关性 describe_analysis.m

```
clc;clear;close;
load data.mat;
class=[1,3,6,9,11,12];
value=[2,4,5,7,8,10,13,14,15,16,17];
max_min_av=[];
R=[];
%% 目标客户对于不同品牌汽车满意度的比较分析
brand1=find(brand==1);
brand2=find(brand==2);
brand3=find(brand==3);

av_brand1=mean(A(brand1,:));
av_brand2=mean(A(brand2,:));
av_brand3=mean(A(brand3,:));
av_score=[av_brand1;av_brand2;av_brand3];
figure(1)
for i=1:size(av_score,2)
    c = categorical({'品牌 1','品牌 2','品牌 3'});
    subplot(2,4,i);
    bar(c,av_score(:,i));
    ylim([min(min(av_score))-1,max(max(av_score))+1])
    title(['a',num2str(i)])
    ylabel('得分');
    grid on
end
B_cls=B(:,class);
B_val=B(:,value);
%% 类别数据 扇形图和众数
for i=1:size(B_cls,2)
    figure(2)
    subplot(2,3,i)
```

```
    bci=B_cls(:,i);
    histogram(bci);
    grid on;
    title(['B',num2str(class(i))])
    xlabel('类别');
    ylabel('频数');
end
%% 数值数据 扇形图和均值
for i=1:size(B_val,2)
    figure(3)
    subplot(3,4,i)
    bvi=B_val(:,i);
    histogram(bvi);
    grid on;
    title(['B',num2str(value(i))])
    xlabel('量值');
    ylabel('频数');

    max_bvi=max(bvi);
    min_bvi=min(bvi);
    av_bvi=mean(bvi);
    max_min_av=[max_min_av;max_bvi,min_bvi,av_bvi];
end

%%数量指标和购买意愿的相关性
B_val(:,5)=2020-B_val(:,5);
%品牌 1
r=[];
B_val_brand1=B_val(brand1,:);
for i=1:size(B_val_brand1,2)
    bvi=B_val_brand1(:,i);
    table=tabulate(will(brand1));
    p=table(2,3)/100;
    q=table(1,3)/100;
    Sx=std(bvi);
    Xp=mean(bvi(find(will(brand1)==1)));
    Xq=mean(bvi(find(will(brand1)==0)));
    r=[r,(Xp-Xq)/Sx*sqrt(p*q)];
end
R=[R;r];
    r=[r;value];
    %相关性排序
    m=sortrows(abs(r'),-1);
    for i=1:size(m,1)
```

```
        if i<size(m,1)
            fprintf('B%d>',m(i,2));
        else
            fprintf('B%d\n',m(i,2));
        end
    end
%品牌 2
r=[];
B_val_brand2=B_val(brand2,:);
for i=1:size(B_val_brand2,2)
    bvi=B_val_brand2(:,i);
    table=tabulate(will(brand2));
    p=table(2,3)/100;
    q=table(1,3)/100;
    Sx=std(bvi);
    Xp=mean(bvi(find(will(brand2)==1)));
    Xq=mean(bvi(find(will(brand2)==0)));
    r=[r,(Xp-Xq)/Sx*sqrt(p*q)];
end
R=[R;r];
    r=[r;value];
    %相关性排序
    m=sortrows(abs(r'),-1);
    for i=1:size(m,1)
        if i<size(m,1)
            fprintf('B%d>',m(i,2));
        else
            fprintf('B%d\n',m(i,2));
        end
    end

 %品牌 3
r=[];
B_val_brand3=B_val(brand3,:);
for i=1:size(B_val_brand3,2)
    bvi=B_val_brand3(:,i);
    table=tabulate(will(brand3));
    p=table(2,3)/100;
    q=table(1,3)/100;
    Sx=std(bvi);
    Xp=mean(bvi(find(will(brand3)==1)));
    Xq=mean(bvi(find(will(brand3)==0)));
    r=[r,(Xp-Xq)/Sx*sqrt(p*q)];
end
```

```
    R=[value;R;r];
    r=[r;value];
    %相关性排序
    m=sortrows(abs(r'),-1);
    for i=1:size(m,1)
        if i<size(m,1)
            fprintf('B%d>',m(i,2));
        else
            fprintf('B%d',m(i,2));
        end
    end

```

第三问 多层感知器预测模型 main 文件:predict_brand1.m

```
clc;clear;close all;
load data.mat;
rand('seed',20);
%% 载入训练集数据 !!!!!!!!数据没有归一化导致
brand1=find(brand==1);
B=[A,B];
will=will(brand1);
B=B(brand1,:);
N_test=7;
index_buy=find(will==1);
index_notbuy=find(will==0);
%打乱顺序 选前 10 个
index_buy=shuffle(index_buy);
index_notbuy=shuffle(index_notbuy);
X_test=[B(index_buy(1:N_test),:);B(index_notbuy(1:N_test),:)]';
Y_test=[will(index_buy(1:N_test),:);will(index_notbuy(1:N_test),:)]';
X_train=[B(index_buy(N_test+1:end),:);B(index_notbuy(N_test+1:end),:)]';
Y_train=[will(index_buy(N_test+1:end),:);will(index_notbuy(N_test+1:end),:)]';
for i=1:size(X_train,1)
    X_train(i,:)=(X_train(i,:)-mean(X_train(i,:)))/sqrt(var(X_train(i,:)));
end
for i=1:size(X_test,1)
    X_test(i,:)=(X_test(i,:)-mean(X_test(i,:)))/sqrt(var(X_test(i,:)));
end

%% 解决训练样本不均衡问题
X_train_extend=X_train;
Y_train_extend=Y_train;
%下面就是重复使用样本 缓解正负样本分布不均的问题
if length(find(Y_train==1))>length(find(Y_train==0))
```

```
    n=length(find(Y_train==1))-length(find(Y_train==0));
    n2=ceil(n/length(find(Y_train==0)));
    for i=1:n2
      X_train_extend=[X_train_extend,X_train(:,find(Y_train==0))];
      Y_train_extend=[Y_train_extend,Y_train(:,find(Y_train==0))];
    end
    n3=length(find(Y_train_extend==0))-length(find(Y_train_extend==1));
    for i=1:n3
      X_train_extend(:,end-i+1)=[];
      Y_train_extend(:,end-i+1)=[];
    end
elseif length(find(Y_train==1))<length(find(Y_train==0))
    n=length(find(Y_train==0))-length(find(Y_train==1));
    n2=ceil(n/length(find(Y_train==1)));
    for i=1:n2
      X_train_extend=[X_train_extend,X_train(:,find(Y_train==1))];
      Y_train_extend=[Y_train_extend,Y_train(:,find(Y_train==1))];
    end
    n3=length(find(Y_train_extend==1))-length(find(Y_train_extend==0));
    for i=1:n3
      X_train_extend(:,end-i+1)=[];
      Y_train_extend(:,end-i+1)=[];
    end
end

X_train=X_train_extend;
Y_train=Y_train_extend;
%% 参数初始化
n_h=[8,8];n_x=25;n_y=1;%%n_h 个数也会影响好坏 !!!
L=2;%一共两层
learning_rate=0.02;%下降学习率  受初值影响很大   学习率太大 会导致 NaN
max_iter = 12000;
iter=0;
cost=[];
acc_train=[];
acc_test=[];
parameters=initialize_parameters(n_h,n_x,n_y);%这个是两层神经网络的初始化函数 多层的要再修改
while iter < max_iter
    cache=forward_propagation(parameters,X_train);
    grad=backward_propagation(parameters,cache,X_train,Y_train);
    parameters = update_parameters(parameters,grad,learning_rate);
        %%记录正确率
    predictions=predict(parameters,X_train);
    err_num=sum(abs(predictions-Y_train));
```

```
    acc_train=[acc_train,1-err_num/length(Y_train)];
    predictions=predict(parameters,X_test);
    err_num=sum(abs(predictions-Y_test));
    acc_test=[acc_test,1-err_num/length(Y_test)];
    iter=iter+1;
end
plot(acc_train,'r','linewidth',1);
hold on;
grid on;
plot(acc_test,'b','linewidth',1);
title('品牌 1');
ylabel('准确率');
xlabel('迭代次数');
legend('训练集准确率迭代曲线','测试集准确率迭代曲线');
fprintf('训练集准确率为%.4f%%\n',acc_train(end)*100);
fprintf('测试集准确率为%.4f%%\n',acc_test(end)*100);

function grad=backward_propagation(parameters,cache,X,Y)
%     使用上述说明搭建反向传播函数。
%     参数：
%      X - 输入数据，维度为（2，数量）
%      Y - “True”标签，维度为（1，数量）
%     返回：
%      dW1 db1 dW2 db2
%
    A3=cache.A3;A2=cache.A2;A1=cache.A1;
    Z3=cache.Z3;Z2=cache.Z2;Z1=cache.Z1;
    W1=parameters.W1;W2=parameters.W2;W3=parameters.W3;
    b1=parameters.b1;b2=parameters.b2;b3=parameters.b3;

    m = length(Y);
    dA3 = - (Y./A3) + (1 - Y)./(1 - A3);

    dZ3 = sigmoid_backward(dA3, Z3);
    [dA2, dW3, db3] = linear_backward(dZ3, A2,W3,b3);

    dZ2 = relu_backward(dA2, Z2);
    [dA1, dW2, db2] = linear_backward(dZ2, A1,W2,b2);

    dZ1 = relu_backward(dA1, Z1);
    [dA0, dW1, db1] = linear_backward(dZ1, X,W1,b1);

    grad.dW1 = dW1;
    grad.dW2 = dW2;
```

```
    grad.dW3 = dW3;
    grad.db1 = db1;
    grad.db2 = db2;
    grad.db3 = db3;
 function cache=forward_propagation(parameters,X)

    [cache.A1,cache.Z1]=linear_activation_forward(X,parameters.W1,parameters.b1,"relu");
    [cache.A2,cache.Z2]=linear_activation_forward(cache.A1,parameters.W2,parameters.b2,"relu")
;
    [cache.A3,cache.Z3]=linear_activation_forward(cache.A2,parameters.W3,parameters.b3,"sigmoi
d");

function cost=compute_cost(AL,Y)

%     实施等式（4）定义的成本函数。
%
%     参数：
%         AL - 与标签预测相对应的概率向量，维度为（1，示例数量）
%         Y - 标签向量（例如：如果是男生，则为0，如果是女生则为1），维度为（1，数量）
%
%     返回：
%         cost - 交叉熵成本
    m=length(Y);
    cost = -sum(log(AL).*Y + log(1 - AL).*(1 - Y)) / m;

function parameters=initialize_parameters(n_h,n_x,n_y)
%n_x 是输入神经元的个数 也就是特征的个数
%n_y 是输出神经元的个数
parameters.W1=(rand(n_h(1),n_x)-0.5)*0.01;
parameters.b1=zeros(n_h(1),1);
parameters.W2=(rand(n_h(2),n_h(1))-0.5)*0.01;
parameters.b2=zeros(n_h(2),1);
parameters.W3=(rand(n_y,n_h(2))-0.5)*0.01;
parameters.b3=zeros(n_y,1);


function [A,Z]=linear_activation_forward(A_prev,W,b,activation)
%     实现 LINEAR-> ACTIVATION 这一层的前向传播
%
%     参数：
%         A_prev - 来自上一层（或输入层）的激活，维度为(上一层的节点数量，示例数）
%         W - 权重矩阵，维度为（当前层的节点数量，前一层的大小）
%         b - 偏向量，维度为（当前层的节点数量，1）
%         activation - 选择在此层中使用的激活函数名，字符串类型，【"sigmoid" | "relu"】
```

```
%
%     返回：
%          A - 激活函数的输出，也称为激活后的值
    if activation == "sigmoid"
        Z=linear_forward(A_prev, W, b);
        A=sigmoid(Z);
    elseif activation == "relu"
        Z=linear_forward(A_prev, W, b);
        A=relu(Z);
    end

function [dA_prev, dW, db]=linear_backward(dZ,A_prev,W,b)
%     为单层实现反向传播的线性部分（第 L 层）
%
%     参数：
%          dZ - 相对于（当前第 l 层的）线性输出的成本梯度
%          cache - 来自当前层前向传播的值的元组（A_prev，W，b）
%
%     返回：
%          dA_prev - 相对于激活（前一层 l-1）的成本梯度，与 A_prev 维度相同
%          dW - 相对于 W（当前层 l）的成本梯度，与 W 的维度相同
%          db - 相对于 b（当前层 l）的成本梯度，与 b 维度相同
    m = size(A_prev,2);%A_prev 为行向量
    dW = dZ*A_prev'/ m;
    db = sum(dZ,2) / m;
    dA_prev = W'*dZ;

function Z=linear_forward(A,W,b)
%     实现前向传播的线性部分。
%
%     参数：
%          A - 来自上一层的激活（或输入数据），维度为(上一层的节点数量，示例的数量）
%          W - 权重矩阵，维度为（当前图层的节点数量，前一图层的节点数量）
%          b - 偏向量，维度为（当前图层节点数量，1）
%
%     返回：
%           Z - 激活功能的输入，也称为预激活参数
    Z=W*A+b;
    function X=normalization(X)
%X 每一行为一个样本
n=size(X,1);%X 中有几个样本
load data.mat;
data=[A,B;X]';%data 每一列为 1 个样本
for i=1:size(data,1)
```

```
205.     data(i,:)=(data(i,:)-mean(data(i,:)))/sqrt(var(data(i,:)));
206. end
207. X=data(:,end-n+1:end);
208. end
209. function predictions=predict(parameters,X)
210.     cache=forward_propagation(parameters,X);
211.     predictions=round(cache.A3);
212. 
213. function A=relu(Z)
214. 
215. %     Implement the backward propagation for a single RELU unit.
216. %
217. %     Arguments:
218. %     dA -- post-activation gradient, of any shape
219. %     cache -- 'Z' where we store for computing backward propagation efficiently
220. %
221. %     Returns:
222. %     dZ -- Gradient of the cost with respect to Z
223. %
224.      A=max(0,Z);
225.      function dZ=relu_backward(dA, Z)
226.     dZ = dA;
227.     dZ(Z<=0) = 0;
228. 
229. function a=shuffle(a)
230.     randIndex = randperm(length(a));
231.     a=a(randIndex);
232. end
233. function A=sigmoid(Z)
234.     A=1./(1+exp(-Z));
235. function dZ=sigmoid_backward(dA, Z)
236. %
237. %     Implement the backward propagation for a single SIGMOID unit.
238. %
239. %     Arguments:
240. %     dA -- post-activation gradient, of any shape
241. %     cache -- 'Z' where we store for computing backward propagation efficiently
242. %
243. %     Returns:
244. %     dZ -- Gradient of the cost with respect to Z
245. %
246.     s = 1./(1+exp(-Z));
247.     dZ = dA .* s .* (1-s);
248. 
```

```
function parameters=update_parameters(parameters,grad,learning_rate)

    parameters.W1 = parameters.W1 - learning_rate * grad.dW1;
    parameters.b1 = parameters.b1 - learning_rate * grad.db1;
    parameters.W2 = parameters.W2 - learning_rate * grad.dW2;
    parameters.b2 = parameters.b2 - learning_rate * grad.db2;
    parameters.W3 = parameters.W3 - learning_rate * grad.dW3;
    parameters.b3 = parameters.b3 - learning_rate * grad.db3;
```

第四问 求解数学规划模型 optimize_brand.m

```
clc;clear;close;
global parameters A B client n;
% choice=1;number=2;
choice=2;number=8;
% choice=3;number=11;
if choice==1
    load para_b1;
elseif choice==2
    load para_b2;
else
    load para_b3;
end
load predict_user.mat;
client=[A(number,:),B(number,:)];
W=130;
% X=zeros(1,8);
% J=object_brand1(X)

A1=ones(1,8);
b=W;
% A1=[];
% b=[];
Aeq=[];
beq=[];
lb=zeros(1,8);
ub=100-A(number,:);
n=0;
x0=rand(1,8);%现在一共 N+3 个参数 w 有 2 个 b1 个 epsilon 有 N 个参数
options = optimoptions('fmincon','Display','iter','Algorithm','sqp');
[x,fval,extiflag]=fmincon(@object_brand,x0,A1,b,Aeq,beq,lb,ub,[],options);
bestX=x;
score=client(1:8)+x;
100*bestX/sum(bestX)
for i =1:length(x)
```

```
    fprintf('第%d 个满意度 %.10f 得分为%.2f 占比%%%.2f\n',i,x(i),score(i),100*x(i)/sum(bestX));
end
fprintf('共需成本%.2f\n',sum(bestX));
x=[client(1:length(x))+x,client(length(x)+1:end)];
x=normalization(x);
cache=forward_propagation(parameters,x);
fprintf('最优值为%.20f\n',cache.A3);

% for i =1:length(x)
%     fprintf('第%d 个满意度占比%% %.10f\n',i,x(i)/W*100);
% end

function J=object_brand1(X)
global parameters client n
X=[client(1:length(X))+X,client(length(X)+1:end)];
X=normalization(X);
cache=forward_propagation(parameters,X);
% J=1/(cache.A3*10^n-4*10^(n-1));
J=1/(cache.A3-0.4999);
%J=1/(cache.A3*10^(5)-floor(cache.A3*10^(5)))*10;
end
```

第四问 遍历 M 值 traverseW.m

```
clc;clear;close;
global parameters A B client n;
% choice=1;number=2;
% choice=2;number=8;
choice=3;number=11;
if choice==1
    load para_b1;
elseif choice==2
    load para_b2;
else
    load para_b3;
end
load predict_user.mat;
client=[A(number,:),B(number,:)];
A1=ones(1,8);
Aeq=[];
beq=[];
lb=zeros(1,8);
ub=100-A(number,:);
n=0;
x0=rand(1,8);%现在一共 N+3 个参数 w 有 2 个 b1 个 epsilon 有 N 个参数
```

```
options = optimoptions('fmincon','Display','iter','Algorithm','sqp');
bestX=[];
best_res=[];
k=-1;
max_iter=40;
cnt=0;
WW=[];
for W=44.7:0.01:45
    WW=[WW,W];
    b=W;
    [x,fval,extiflag]=fmincon(@object_brand,x0,A1,b,Aeq,beq,lb,ub,[]);
    bestX=[bestX;x];
    S=sum(x);
    x=[client(1:length(x))+x,client(length(x)+1:end)];
    x=normalization(x);
    cache=forward_propagation(parameters,x);
    fprintf('W=%d 时,最优值为%.10f，共耗费%.2f \n',W,cache.A3,S);
    if cache.A3>0.5&&k==-1
       W0=W;%W0 是临界值
       res0=cache.A3;
       fprintf('!!!!!!!!!!!!!!!!!!!!当 W=%d 时,顾客开始愿意购买汽车!!!!!!!!!!!!!!!!!!!!!!!!!',W);
       k=0;
    end
    best_res=[best_res,cache.A3];
    if S+1<W
        cnt=cnt+1;
    end
    if cnt>max_iter
        Smax=S;
        break
    end
end

plot(WW,best_res,'b-','linewidth',1.5);
hold on;
grid on;
plot(W0,res0,'.r','markersize',20);
plot([WW(1),WW(end)],[0.5 0.5],'--b','linewidth',0.5);
% plot([WW(1),WW(end)],[Smax Smax],'--o','linewidth',1.5);
text(W0+2,res0,['临界点 W=',num2str(W0)],'FontSize',14,'Color','red');
ylabel('购买意愿');
xlabel('W');
% score=client(1:8)+x;
% for i =1:length(x)
```

```
%     fprintf('第%d 个满意度 %.10f 得分为%.2f \n',i,x(i),score(i));
% end
% fprintf('共需成本%.2f\n',sum(bestX));
% x=[client(1:length(x))+x,client(length(x)+1:end)];
% x=normalization(x);
% cache=forward_propagation(parameters,x);
% fprintf('最优值为%.20f\n',cache.A3);

% for i =1:length(x)
%     fprintf('第%d 个满意度占比%% %.10f\n',i,x(i)/W*100);
% end
```